



Aula 1 - Minicurso Ruby on Rails

Andrei Paes Figueiredo





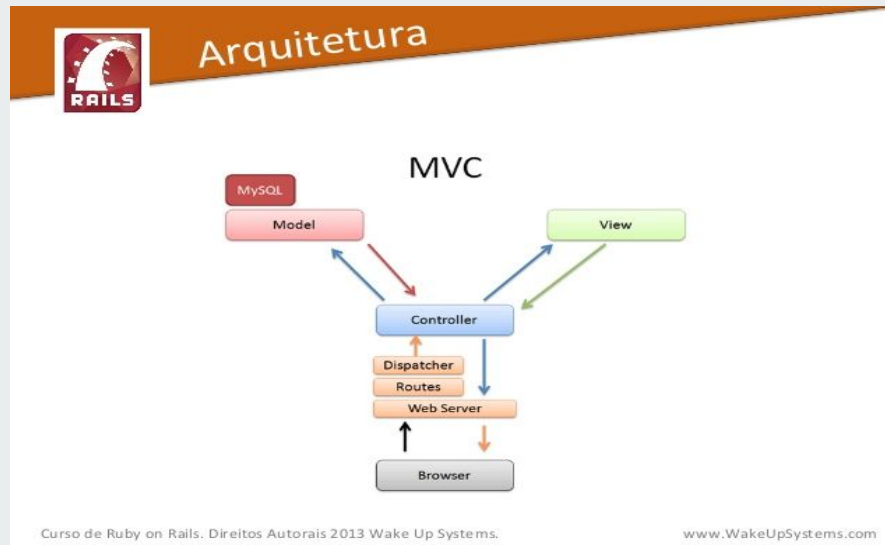
História do Rails

- Webdevelopment Framework
- Open Source
- Felicidade do Programador
- Arquitetura Model-View-Controller
- Baseado em generators



Model-View-Controller

- Isolar partes do projeto
- Facilitar manutenção do código





Generators

- Automatizar criação de arquivos e pastas
- Acelera desenvolvimento
- Evita erros

- rails new appminicurso





Estrutura de Pastas

FOLDERS

▼ seeyouaround

▼ app

- ▶ assets
- ▶ controllers
- ▶ helpers
- ▶ mailers
- ▶ models
- ▶ views

▶ bin

▶ config

▶ db

▶ lib

▶ log

▶ public

▶ test

▶ tmp

▶ vendor



Bundle

- Gerenciado das gems
- Reutilização de gems





Domain Driven Development

- Aumentar a vida útil da aplicação
- Facilita o desenvolvimento da aplicação
- Não costuma ser usado em aplicações Rails
- Tem sido incentivado o uso com o passar dos anos





Full Stack

- Contem ferramentas tanto para front quanto back-end
- Desenvolvedor pode trabalhar com a mesma competencia em ambos





Documentação

- <http://ruby-doc.org/>
- <https://apidock.com/ruby>
- <http://tryruby.org/levels/1/challenges/0>
- <https://www.jmonteiro.com/aprendaaprogramar/>






Irb

- Console do ruby
- Só escrever e começar a programar em ruby



- 
- Array
 - Hash
 - Métodos
 - Parâmetros
 - Blocos



- 
- Classe - Tudo é objeto!
 - Módulos
 - Coleção de Métodos e constantes
 - Elimina a necessidade de herança múltipla





Bibliografia

- <https://code.tutsplus.com/articles/ruby-on-rails-study-guide-the-history-of-rails--net-29439>
- <http://fabiotionio.com.br/blog/2009/01/17/entendo-a-estrutura-de-diretorios-do-rails/>

